



Definiendo el modelo

Transcripción

[00:00] Hola a todos y todas, bienvenidos y bienvenidas una vez más este curso de Servlets. Ya en el video pasado vimos cómo hacer para que un Servlet reciba específicamente datos con un método POST y no un GET. Y ya vamos avanzando dentro de nuestro Servlet.

[00:22] Ahora lo que me gustaría hacer es nuestro Servlet lo que tiene que hacer es registrar empresas. ¿Cómo tiene que registrar una empresa? Sería bueno comenzar a trabajar en el modelo de nuestra aplicación. Ese modelo, como nosotros tenemos que registrar empresas, sería bueno crear una clase de empresa que represente a cada una de esas posibles empresas que vamos a agregar.

[00:47] Entonces para comenzar con eso vamos a ir a nuestro paquete y vamos a hacer clic derecho, nuevo new class y acá vamos a crear una clase llamada empresa. Con esa clase, nosotros vamos ahora a agregar algunos atributos. Primero y principal vamos a agregar el atributo del tipo string llamado nombre.

[01:11] Nuestra empresa tiene que tener un nombre. Y ahora ya que vamos a trabajar con una base de datos, vamos a guardar empresas, no vamos a trabajar exactamente con una base de datos verdadera, pero vamos a simular una base de datos. ¿Entonces cómo vamos a hacer eso? Sería bueno que nuestra empresa tenga un id para ser identificado unívocamente dentro de la base de datos.

[01:35] Entonces para eso vamos a hacer un `private Integer id`. Coloco `Integer` porque tiene algunos métodos más. Bien, ahora acá en un espacio en blanco, voy a crear los getters y setters de esos atributos. Para hacer eso, vamos a ir a clic derecho source y generate getters and setters. Vamos a hacer clic en estos cuadraditos de `id` y nombre, porque queremos todos esos getters y setters y vamos a hacer clic en generate.

[02:05] Vamos a dar algún espacio acá para que quede mejor, y bien, ya tenemos nuestra empresa. Ahora vamos a ir a nuestro Servlet y como tenemos nuestra clase empresa, voy a guardar acá para que no se me olvide, vamos a crear esa empresa, ahora que en la request nosotros recibimos un nombre de una empresa, sería bueno ya comenzar a crear esa empresa.

[02:30] Entonces para eso vamos a hacer `empresa empresa = new Empresa()` podría haber hecho que en el constructor reciba un parámetro que es el nombre, pero bueno, ya que lo hemos hecho así, vamos a continuar. Vamos a hacer `empresa.setNombre` y `nombreEmpresa`, que es el atributo que hemos recibido en la request. Perfecto. Ya tenemos una empresa.

[02:55] Ahora lo que tendríamos que hacer es simular una base de datos, ya que como dije antes no vamos a usar una base de datos verdadera porque haría que el curso sea mucho más extenso. Entonces vamos a simplemente simular una base de datos, ¿cómo hacemos eso? Vamos a comenzar creando una clase que simule esa base de datos. Entonces vamos a crear la clase DB.

[03:17] Voy a llamarla `baseDeDatos = new DB();` Eclipse nos está diciendo: no existe esa clase. Entonces nos da la opción de crear esa clase, entonces vamos a hacer clic en create class solamente colocando el mouse arriba de esa clase y ya nos permite crear esa clase. Vamos a dar en finish y acá tenemos nuestra base de datos.

[03:47] Bien. Una vez que tenemos una base de datos, nosotros queremos agregar una empresa nueva a esa base de datos, para eso vamos a hacer

`baseDeDatos.agregarEmpresa` y vamos a enviar como parámetro la empresa. Bien. Ustedes dirán: “pero no existe ese `agregarEmpresa`”. Bueno. Eclipse nos dice: “no existe ese método `agregarEmpresa`”. Entonces nosotros podemos crear ese método.

[04:14] Perfecto. Ya Eclipse nos creó el método que recibe una empresa. Y ahora viene una parte complicada. ¿Cómo hacemos? Fíjense que cada vez que nosotros llamamos a ese POST, nosotros estamos creando una nueva base de datos. Entonces, si nosotros guardamos una empresa en un objeto de tipo base de datos, nosotros estaríamos guardando una empresa nueva en objetos diferentes.

[04:43] Y no es lo que queremos. Nosotros queremos mantener cada una de esas empresas que nosotros agregamos a la base de datos, entonces vamos a usar un pequeño truquito acá que es usar variables estáticas. Vamos a crear una variable estática acá en la base de datos que va a ser de la clase y no de los objetos, entonces todos los objetos van a compartir el mismo atributo.

[05:07] Entonces todos van a poder acceder a una lista de las empresas, entonces para crear esa lista de empresas vamos a comenzar con `private static`, muy importante, del tipo `List`, ese `List`, esperen que tengo que importarlo con “Ctrl + espacio” y tiene que importar ese `java.util`.

[05:30] No importen otro `list` que tengan por ahí en la lista, porque ese es el que vamos a utilizar. Doble clic y vamos a colocar cuál tipo de datos van a estar dentro de esa lista. En ese caso, vamos a guardar empresas, entonces empresas. Y acá el nombre le vamos a colocar `lista empresas` igual a `new` y acá vamos a especificar cuál es el tipo de la lista, vamos a decir que es un `ArrayList`.

[06:00] `ArrayList`, bien, acá vamos a sacar eso porque ya este `generics` de empresas ya nos está este `ArrayList` va a heredar ese `generic` y va a decir que es del tipo empresas. Acá nos está diciendo que perdón, coloqué empresas, es empresa. Va a ser una lista de empresas. Bien, tenemos esa lista.

[06:26] Ahora, cuando nosotros queremos agregar una empresa que llama este método de `agregarEmpresa`, lo que nosotros vamos a hacer es `listaEmpresas.add` y vamos a agregar esa empresa que hemos recibido como atributo. Perfecto. Y una vez que hemos agregado la empresa, nosotros también queremos un método para poder ver esas empresas, entonces vamos a crear un método acá `public` que va a devolver una lista de empresas y vamos a llamarlo de `getEmpresas`.

[07:07] No recibe ningún parámetro y va a hacer un `return` de `listaEmpresas`. Con eso ya estaríamos, y una buena idea sería especificar cuál `listaEmpresas` vamos a devolver. En este caso existe solamente una pero es bueno siempre especificar y decir `DB.ListaEmpresas`. O sea el atributo `listaEmpresas` de nuestra clase `DB`.

[07:40] Entonces hacemos `DB.ListaEmpresas` acá también y perfecto, ya tenemos nuestra base de datos simulada y ya estaríamos guardando una empresa en esa base de datos. Bien, vamos a probar, a ver si está funcionando todo bien, si no hay ningún problema.

[08:00] La consola, no hay ningún error. Bien, vamos a ir acá a nuestro `formNuevaEmpresa`, actualizar y vamos a colocar Alura, enviar, empresa Alura registrada! No ha habido ningún error. Si nosotros nos vamos acá a la consola, no hay ningún error. Bien. Perfecto, no ha cambiado nada, o sea, desde el punto de vista de Google Chrome no hay ninguna modificación.

[08:27] Esto es todo lo que hemos hecho es todo por dentro. Perfecto, ya hemos podido simular una base de datos donde agregamos una empresa, tenemos nuestra clase empresa que representa cada una de las empresas que nosotros agregamos, y con esto ya vamos un paso más adelante a nuestro objetivo final. Entonces hagan los ejercicios y los espero en el próximo video.

